

Kent Academic Repository

Full text document (pdf)

Citation for published version

Moon, Ilkyeong and Salhi, Said and Feng, Xuehao (2020) The location-routing problem with multi-compartment and multi-trip: formulation and heuristic approaches. *Transportmetrica A: Transport Science*, 16 (3). pp. 501-528. ISSN 2324-9935.

DOI

<https://doi.org/10.1080/23249935.2020.1720036>

Link to record in KAR

<https://kar.kent.ac.uk/80010/>

Document Version

Author's Accepted Manuscript

Copyright & reuse

Content in the Kent Academic Repository is made available for research purposes. Unless otherwise stated all content is protected by copyright and in the absence of an open licence (eg Creative Commons), permissions for further reuse of content should be sought from the publisher, author or other copyright holder.

Versions of research

The version in the Kent Academic Repository may differ from the final published version.

Users are advised to check <http://kar.kent.ac.uk> for the status of the paper. **Users should always cite the published version of record.**

Enquiries

For any further enquiries regarding the licence status of this document, please contact:

researchsupport@kent.ac.uk

If you believe this document infringes copyright then please contact the KAR admin team with the take-down information provided at <http://kar.kent.ac.uk/contact.html>

The Location Routing Problem with Multi-Compartment and Multi-Trip: Formulation and Heuristic Approaches

ILKYEONG MOON¹, SAID SALHI² and XUEHAO FENG^{3*}

¹ Department of Industrial Engineering and Institute for Industrial Systems Innovation, Seoul National University, Seoul 08826, KOREA

² Centre for Logistics and Heuristic Optimization, Kent Business School, University of Kent, Canterbury CT2 7PE, United Kingdom

³ Ocean College, Zhejiang University, Hangzhou 310058, China

Abstract

The location routing problem with multi-compartment and multi-trip is an extension to the standard location routing problem. In this problem, depots are used to deliver different products using heterogeneous vehicles with several compartments. Each compartment has a limited capacity and is dedicated to a single type of product. The problem is formulated as a mixed integer program. A constructive heuristic and a hybrid genetic algorithm (HGA) are proposed. Numerical experiments show that both heuristics can efficiently determine the optimal solutions on small size instances. For larger ones, the HGA outperforms the constructive heuristic with relatively more computational time. Managerial insights have been obtained from sensitivity analyses which would be helpful to improve the performance of the supply network.

Keywords: location routing problem; multi-compartment; multi-trip; mixed integer program; heuristic.

*Corresponding author: Dr. XUEHAO FENG

Address: Ocean College, Zhejiang University, Hangzhou 310058, CHINA

Tel: +86-15858213431

E-mail: fengxuehao@zju.edu.cn

1. Introduction

The success of many organizations depends on the location of their facilities (plants, warehouses, distribution centers, among others) and their routing plans. These two logistical problems constitute crucial choices among strategic, tactical and operational decisions (Perl, 1983; Zarandi, 2011; Wang *et al.*, 2013). It has also been shown empirically that suboptimal solutions can be obtained if the two levels of decisions are tackled separated (Salhi and Rand, 1989). To overcome these drawbacks, the classical location routing problem (LRP), a combination of both the depot location and the routing decisions, has been well studied in the literature. However, in this study which we will define later, we need to relax two assumptions of the LRP which include (i) all types of products are loaded in one compartment; and (ii) each vehicle is allowed to make at most one trip during its workday. In other words, our problem requires the use of more than one compartment and that multiple trips are allowed to be used due to the short round trips.

The problem presented in this paper involves a typical food company that needs to use several depots at different locations in a distribution network. Through the depots, the company replenishes groceries using heterogeneous vehicles with multiple compartments, each of which is exclusive for a specific type of food. This model reflects many real food transport networks in which food types are categorized into two main groups, frozen and unfrozen, which requires truck compartments dedicated to only frozen or only unfrozen products, respectively. In addition, for some cases, each truck is equipped with more than two compartments; for example, some vegetables, such as onions, should be loaded in a dedicated compartment because the odor will taint other deliverables, such as cooked food or prepared snacks. The groceries are expected to have less-than-truckload demands, which are replenished from routes that involve multiple stops. Given that most of round trips can be performed in a short time period, each vehicle can make multiple trips to achieve cost savings due to a reduced number of vehicles and drivers. There is also a maximum distance allowed for each vehicle during a workday. Hence, the problem addressed in this paper is a location routing problem with multi-compartment and multi-trip. To minimize the total daily transportation cost, the decision makers need to simultaneously determine the number of depots, their respective locations, the assignment of customers to these open depots, the deployment of vehicles, the route configuration for every vehicle, and the assignment of products to the vehicles.

The proposed location routing problem can be regarded as the extension to three problems which include (i) the classical location routing problem (LRP), (ii) the vehicle routing problem with multi-compartment (VRPMC), and (iii) the vehicle routing problem with multi-trip (VRPMT). We refer to this LRP with Multiple Compartments and Multiple Trips as LRPMCMT for short. Table 1 summarizes the main features of the four problems including ours. For clarity of presentation, we briefly review (i)-(iii) as their combination makes up this new logistical problem which we are aiming to address.

[Insert Table 1 about here]

(i) A brief review on the LRP

If multiple compartments and multiple trips are not considered, then the combined problem will be reduced to the classical LRP. Min *et al.* (1998) provided a survey of the literature on location-routing up to the late 1990s; this work explored promising research opportunities in terms of the incorporation of more realistic aspects, algorithmic design, and model complexity. The paper showed that most early published papers did not consider capacitated routes and capacitated depots simultaneously. Nagy and Salhi (2007) provided a survey of LRPs, proposed a classification scheme, introduced a number of problem variants and provided a summary of LRP applications such as those found in the food and drink distribution (e.g., Watson-Gandy and Dohrn, 1973) and waste collection (e.g., Kulcar, 1996). Since then, interesting studies were produced. For instance, Berger *et al.* (2007) presented a set-partitioning-based formulation for uncapacitated location-routing problems with distance constraints. Their model can be applied to perishable goods delivery, time critical delivery, and other areas. A branch-and-price algorithm was developed to solve the problem. The LRP with both capacitated depots and routes was studied by various researchers including Prins *et al.* (2006, 2007); Duhamel (2010) and Yu *et al.* (2010). The LRP has received relatively less attention when compared to other extensions of the VRP, and more specifically only a few extensions to the LRP were addressed in the literature. Among these Albareda-Sambola *et al.* (2007) introduced a stochastic location-routing problem in which customer demand was stochastic so that only a subset of the customers required service after a decision was made. Prodhon (2007) combined the periodic VRP and LRP into an even more realistic problem termed the periodic LRP, in which periodic routing decisions were considered during a time horizon. A simple iterative heuristic was proposed to tackle the problem. Later, the same problem was solved by Prodhon and Prins (2008) using a multi-start genetic-based metaheuristic and a hybrid evolutionary algorithm respectively. Zarandi *et al.* (2011) addressed the fuzzy version of the capacitated location routing problem, in which the travel time between two nodes is considered as a fuzzy variable. They proposed a simulation-embedded simulated annealing procedure to solve the problem. Very recently, an interesting and comprehensive survey on LRP is given by Prodhon and Prins (2014) who reviewed the new variants and techniques developed since the Nagy and Salhi's last review. The authors also highlighted useful suggestions for the future including the design of exact methods that incorporate problem structures, realistic problems where customers are not all served, among others. Liu *et al.* (2016) analyzed the dynamic activity-travel assignment with transport can location capacity constraints. In that study, the dynamic traffic assignment is combined with the activity-based modelling in a unified framework. Menezes *et al.* (2016) developed a model for an LRP that involves determining the optimal location of regional distribution centers and designing the routes of the trucks that undertake the transportation among such centers and demand nodes. Chen *et al.* (2017) discussed the design of suburban bus routing problem for

airport access, in which the pickup locations are selected from candidate stops. That study implemented an artificial bee colony approach to obtain solutions of good quality.

To the best of our knowledge, the LRP with multi-compartments with the presence of multi-trips has not been studied in the above applications. This variant constitutes an important extension to the classical LRP that has not been investigated in the literature and for which we believe it is worthwhile exploring.

(ii) A brief review on the VRPMC

The vehicle routing problem with multi-compartment (VRPMC) is a special case of the LRPMTMC in that the depot locations have been determined and a vehicle can make at most one trip during one workday. There are a few papers dealing with this particular routing problem. This kind of studies focuses on two different categories based on whether the compartment is dedicated to the product or not. Fuel distribution is a well-known application of the latter category, in which one compartment can load any single type of fuel. Interesting practical applications can be found in Brown and Graves (1981) and Bruggen *et al.* (1995). Our study focuses on the former strategy where one compartment is dedicated to one product only which is applicable to food distribution and waste collection. Chajakis and Guignard (2003) introduced logical constraints to tighten their two integer programming models when scheduling food deliveries to convenience stores. Fallahi *et al.* (2008) studied the distribution of cattle food to farms in the West of France. A constructed heuristic, a memetic algorithm, and a tabu search were proposed to solve the problem. Muyldermans and Pang (2010) studied the benefits of co-collection using vehicles with multiple compartments through a comparison with separate collections using traditional vehicles. A competitive local search procedure was proposed to solve the problem and sensitivity analysis was conducted to consider different factors. Silvestrin and Ritt (2017) proposed an efficient tabu search algorithm to solve the problem. Mendoza *et al.* (2010) proposed a new memetic algorithm to solve the VRPMC with stochastic demands. Later, Mendoza *et al.* (2011) introduced three new constructive heuristics to expand the existing tool box that is used for solving the problem. Various applications on the VRPMC are provided by Derigs *et al.* (2011) who also presented a portfolio of the competitive components that were found to be competitive for solving the routing part of the combined problem.

(iii) A brief review on the VRPMT

When depot locations have been determined and multiple compartments are not considered, the problem becomes the vehicle routing problem with multi-trip (VRPMT). This logistical problem commonly exists in practice as vehicles are used to make multiple trips during one workday. The first study involving this problem is presented by Fleischmann (1990). In his paper, a bin packing heuristic was used to assign routes to vehicles. To solve the problem, Taillard *et al.* (1996) proposed a three-phase approach where a bin packing algorithm was used for the assignment of routes to vehicles in the second phase. Brandao and Mercer (1998) proposed a tabu

search heuristic without resorting to a multi-phase methodology. Prins (2002) studied the VRPMT with the presence of limited heterogeneous fleet. Later, Petch and Salhi (2004) developed a multi-phase constructive heuristic whereas Salhi and Petch (2007) designed a GA. Olivera and Viera (2007) provided an adaptive memory approach. Battarra *et al.* (2009) and Azi *et al.* (2010) analyzed the VRPMT with time windows.

The contribution of this study is threefold. (i) we aim to address for the first time in the literature the logistical problem namely the LRPMCMT problem, (ii) we present a comprehensive mixed integer formulation which can be used as a basis for further tightening if necessary as well as one of the performance measures for the assessment of the heuristics on small size instances, and finally (iii) we design a constructive heuristic and a hybrid genetic algorithm to efficiently solve large size instances.

The rest of the paper is organized as follows. Section 2 presents the mathematical model. Sections 3 and 4 provide a constructive heuristic and a hybrid genetic algorithm respectively. Computational experiments are given in Section 5. The final section summarizes our findings and provides some highlights for future research.

2. Mathematical Model

The location routing problem with multi-compartment and multi-trip is defined as follows: In the distribution network, there is a supplier and a set of customers that order multiple types of products from the supplier. The supplier can divide all types of products into some main types, according to the exclusive use of certain types of food (e.g., frozen and non-frozen). Without loss of generality, in this model, we consider vehicles with the same number of compartments but with different capacities and travelling distance limitations. Each type of product has a one-to-one correspondence with each type of compartment. The supplier needs to select a number of uncapacitated depots from the set of potential locations. In the problem, the following exogenous parameters are given:

- The location and demand of every customer.
- The location and the fixed cost of each potential depot.
- The capacity of every compartment of the heterogeneous vehicles.
- The fixed deployment cost and the maximum distance allowed for every vehicle.
- The unit running cost per kilometer for every vehicle.

To minimize the total daily transportation cost, the problem simultaneously involves determining the number of depots and their locations, the assignment of customers to these open depots, the deployment of vehicles, the routes for every vehicle, and the assignment of products to the vehicles. The assumptions used in this study include:

- Each customer has a non-zero demand for at least one type of product.
- All demands should be satisfied.
- The vehicles are initially located at the depots.
- Each vehicle must return to the same depot from which it departs.

- Each compartment is dedicated to one type of product.
- Each vehicle can make multiple trips if necessary.
- Each type of product of each customer is delivered on one trip by one vehicle.
- Each depot is large enough to cater for all products.
- There is an unlimited number of vehicles available to deliver the products.

The following notation is used in the model.

Sets (Indices)

I = set of potential depots ($i \in I$).

J = set of customers ($j \in J$).

$U = I \cup J$ = set of all nodes in the network (customers and potential sites)

T = set of trips, where $|T|$ is large enough to accommodate the maximal number of trips that the fleet can possibly make, ($t \in T$).

V = set of vehicles ($v \in V$).

P = set of products ($p \in P$).

Parameters

L_{ij} = distance (in kilometers) between nodes i and j ($i, j \in U$).

D_{jp} = demand of customer j for product p ($p \in P, j \in J$).

Q_{vp} = capacity of compartment p in vehicle v ($v \in V, p \in P$).

F_v = fixed cost (in \$) of using vehicle v ($v \in V$).

C_v = transportation cost (in \$) per kilometer for vehicle v ($v \in V$).

G_i = amortized fixed cost (in \$) of establishing a depot at site i ($i \in I$).

ML_v = maximum distance that could be travelled by vehicle v ($v \in V$).

$N = |J|$, number of customers.

M = a very large number (say $M = N^2$).

Decision variables

$$x_{ijt} = \begin{cases} 1 & \text{if vehicle } v \text{ travels from node } i \text{ to node } j \text{ during trip } t. \\ 0 & \text{otherwise.} \end{cases}$$

$$y_{jptv} = \begin{cases} 1 & \text{if vehicle } v \text{ delivers product } p \text{ to customer } j \text{ during trip } t \\ 0 & \text{otherwise} \end{cases}$$

$$l_v = \begin{cases} 1 & \text{if vehicle } v \text{ is used} \\ 0 & \text{otherwise} \end{cases}$$

$$w_i = \begin{cases} 1 & \text{if depot } i \text{ is used} \\ 0 & \text{otherwise} \end{cases}$$

$$s_{ij} = \begin{cases} 1 & \text{if customer } j \text{ is assigned to depot } i \\ 0 & \text{Otherwise} \end{cases}$$

$$z_{vi} = \begin{cases} 1 & \text{if vehicle } v \text{ is assigned to depot } i. \\ 0 & \text{otherwise} \end{cases}$$

A_{mtv} is the auxiliary variable for sub-tour elimination constraints for trip t of vehicle v visiting customer m .

Formulation

Minimize

$$\sum_{i \in I} G_i w_i + \sum_{i \in U} \sum_{j \in U} \sum_{t \in T} \sum_{v \in V} C_v L_{ij} x_{ijtv} + \sum_{v \in V} F_v l_v \quad (1)$$

Subject to

$$\sum_{t \in T} \sum_{v \in V} y_{jptv} = 1, \forall j \in J, \forall p \in P, \quad (2)$$

$$\sum_{j \in J} y_{jptv} D_{jp} \leq Q_{vp}, \forall p \in P, \forall t \in T, \forall v \in V, \quad (3)$$

$$y_{jptv} \leq \sum_{i \in U} x_{ijtv}, \forall j \in J, \forall p \in P, \forall t \in T, \forall v \in V, \quad (4)$$

$$\sum_{j \in U} x_{ijtv} - \sum_{j \in U} x_{jivt} = 0, \forall i \in U, \forall t \in T, \forall v \in V, \quad (5)$$

$$\sum_{j \in J} s_{ij} \leq M w_i, \forall i \in I, \quad (6)$$

$$-s_{ij} + \sum_{u \in U} (x_{iuv} + x_{ujiv}) \leq 1, \forall i \in I, \forall j \in J, \forall t \in T, \forall v \in V, \quad (7)$$

$$\sum_{i \in I} z_{vi} \leq l_v, \quad \forall v \in V, \quad (8)$$

$$\sum_{j \in J} \sum_{t \in T} x_{ijtv} \leq Mz_{vi}, \quad \forall i \in I, \forall v \in V, \quad (9)$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijtv} \leq 1, \quad \forall v \in V, \forall t \in T, \quad (10)$$

$$A_{mtv} - A_{jtv} + Nx_{mjtv} \leq N - 1, \quad \forall m, j \in J, \forall t \in T, \forall v \in V, \quad (11)$$

$$\sum_{i \in U} \sum_{j \in U} \sum_{t \in T} x_{ijtv} L_{ij} \leq ML_v, \quad \forall v \in V, \quad (12)$$

$$\begin{aligned} x_{ijtv} &\in \{0,1\}, \quad \forall i \in U, \forall j \in U, \forall t \in T, \forall v \in V \\ y_{jptv} &\in \{0,1\}, \quad \forall j \in J, \forall p \in P, \forall t \in T, \forall v \in V \\ w_i &\in \{0,1\}, \quad \forall i \in I \\ l_v &\in \{0,1\}, \quad \forall v \in V \\ s_{ij} &\in \{0,1\}, \quad \forall i \in I, \forall j \in J \\ z_{vi} &\in \{0,1\}, \quad \forall v \in V, \forall i \in I \end{aligned} \quad (13)$$

$$A_{mtv} \geq 0, \quad \forall m \in J, \forall t \in T, \forall v \in V \quad (14)$$

The objective function (1) is to minimize the summation of the fixed costs of using the depots (the first term), the transportation costs (the second term), and the fixed charges of using the vehicles (the third term). We developed the decision variable y_{jptv} and Constraints (2) to (4) to incorporate multi-compartment and multi-trip simultaneously in the LRP model. Constraint (2) guarantees that each product ordered by a customer is delivered by one vehicle during one trip and Constraint (3) is a capacity constraint that is set for the compartments. Constraint (2) does not imply that each customer must order all types of products. Because we assume that each customer has a non-zero demand for at least one type of product, in any feasible solution, at least one y_{jptv} is equal to 1 for all $\{j,p,t,v\}$. For instance, $D_{21}=10$ and $D_{22}=0$ means that Customer 2 has ordered 10 units of Type 1 products and has no demand for any Type 2 products. In this case, the constraint ensures that $\sum_{t \in T} \sum_{v \in V} y_{21tv} = 1$; i.e., in one trip, Type 2 products must be delivered to Customer 1. We

also set $y_{2111}=1$ in the optimal solution, which means that Vehicle 1 delivers Type 1 products to Customer 2 on the first trip of one day. Then, we set $y_{2211}=1$ in the optimal solution without increasing the value of the objective function from that found when $y_{2211}=0$. In this case, Constraints (2) and (3) are also satisfied when $j=2$ and $p=2$, even though $D_{22}=0$. Therefore, we found that Constraint (2) does not require non-zero demand for all types of product by each customer. Constraint (4) ensures that vehicle v can deliver product p to customer j during trip t only if customer j is visited by vehicle v during trip t . This constraint links the multi-compartment consideration with the multi-trip route construction issue. The involvement of y_{jptv} is associated with the

$|J| \cdot |P| \cdot |T| \cdot |V|$ decision variables, and taking multiple compartments into account, we obtained $|P| \cdot [(1+|T| \cdot |V|)(1+|J|)-1]$ constraints.

Constraint (5) guarantees that if one node is visited by one vehicle in one trip, that vehicle should depart from this node. Meanwhile, if one node is not visited by any vehicle in any trip, no trip can start from that node. This constraint is also known as *flow conservation*. Constraint (6) specifies that customers can be assigned to a depot only if the depot is used. For Constraint (7), $\sum_{u \in U} (x_{iuv} + x_{ujv})$ should be no greater than

1 for any trip t of vehicle v if $s_{ij}=0$. In this case, customer j and depot i should not be on the same route at the same time. Hence, this constraint guarantees that a depot can serve a customer via a trip only if that customer is assigned to that depot. Using the same idea, we can obtain the following two constraints: Constraint (8) specifies that each vehicle can be assigned to, at most, one depot, and Constraint (9) ensures that vehicle v can depart from depot i only if vehicle v is assigned to depot i . Because Constraint (9) cannot avoid the infeasible case in which one vehicle directly visits more than one customer from one depot during one trip, we propose Constraint (10) to ensure that any trip for each vehicle can directly connect no more than one pair consisting of a depot and a customer. Constraint (11) features a set of sub-tour elimination constraints that make all trips without any depots unsatisfactory for use. In conjunction with Constraint (5), these three constraints guarantee that any trip of any vehicle must lead to a feasible solution of a TSP problem with one depot and assigned customers. Fig. 1 illustrates the idea of Constraint (11) with an example such that Customers 2 - 5 are assigned to Depot 1. As Fig. 1 shows, one trip for one vehicle (e.g., vehicle v and trip t) covers two routes. Then, we obtained $x_{12tv}=x_{21tv}=1$, and $x_{34tv}=x_{45tv}=x_{53tv}=1$. This solution satisfies Constraints (5), (9), and (10), but it is not a feasible solution in the real world. Likewise, it will not be obtained in our model because of Constraint (11). From this constraint, we have

$$\begin{cases} A_{3tv} - A_{4tv} + Nx_{34tv} \leq N - 1 & (15) \\ A_{4tv} - A_{5tv} + Nx_{45tv} \leq N - 1 & (16) \\ A_{5tv} - A_{3tv} + Nx_{53tv} \leq N - 1 & (17) \end{cases}$$

By summing Inequalities (15) - (17), we can obtain $3N \leq 3N - 1$, which implies that no $\{A_{3tv}, A_{4tv}, A_{5tv}\}$ can make $x_{34tv}=x_{45tv}=x_{53tv}=1$ feasible. We can also easily show that Constraint (11) is always satisfied if a solution does not contain any sub-tour. Constraint (12) offers the distance constraint set for each vehicle. Constraint (13) includes the binary requirements for the variables, and Constraint (14) features non-negative auxiliary variables. In our model, any customer's demand for one type of product should be satisfied by one vehicle during one trip. In addition, this model allows different vehicles to satisfy one customer's demand for different types of products, which provides greater flexibility in routing for the multi-product case.

This problem is a mixed 0-1 linear programming problem that contains $|J| \cdot |T| \cdot |V|$ continuous variables, $|T| \cdot |V|(|U|^2 + |J| \cdot |P|) + |V|(1+|I|) + |I|(1+|J|)$ binary variables, and $|J| \cdot |T| \cdot |V|(|J| + |I| + |P|) + |V|[2+|I|+|T|(1+|P|+|U|)] + |I| + |P| \cdot |J|$ constraints without considering Constraints (13) and (14). The consideration of multiple trips influences the

complexity of the model in terms of decision variables and constraints. First, the decision variable, y_{jptv} is involved, and the number of x_{ijtv} and A_{mtv} are also influenced. Second, the number of constraints related to the LRP is multiplied by $|T|$. In addition, Constraint (4) is specified for this consideration.

[Insert Figure 1 about here]

As explained, the problem addressed here is NP hard because it can be regarded as an extension to the following three problems: an LRP, VRPMC, and VRPMT, which are all known to be NP hard. The mathematical formulation, therefore, is appropriate to use for small instances in which the results are used only for evaluating the performance of the heuristics. The only way to address instances of moderate size is to use heuristics and meta-heuristics.

The LRP with the limitation in travelling distances and capacitated vehicles may involve a balance between the transportation cost and the fixed cost of vehicles. For example, the transportation cost presented in Figure 2(a) is lower than that shown in Figure 2(b). However, the vehicle cannot visit Customers 3 - 6 in one trip because of the compartment capacity or maximum travelling distance limitation. For either of these two reasons, two routes are required: {Depot 2→Customer 4→Customer 3→Depot 2} and {Depot 2→Customer 6→Customer 5→Depot 2}. If the total travelling distance extends beyond the limitation of a vehicle, an extra vehicle is employed to finish one of the trips, which leads to a higher fixed cost. In the scenario featured in Figure 2(b), Customer 3 receives delivery from Depot 1, and then, one vehicle can visit Customers 4 - 6 in one trip. In this case, the fixed cost of the vehicle is decreased, while the transportation cost is increased. This trade-off could be more significant if each customer demands multiple types of products, which is a novel characteristic of the LRPMCMT. In this study, we propose a constructive heuristic followed by a hybrid genetic algorithm, both of which are described further in the following sections.

[Insert Figure 2 about here]

3. A Constructive-Based Heuristic

For this heuristic algorithm, we first selected depots and construct routes by tailored algorithms. In particular, we considered the utilization ratios of compartment capacities and the maximum travelling distances of the vehicles. Then, we developed a Packing-Strategy-based-Local-Search Procedure that reconstructs obtained routes after an “insertion” or “exchange” to see any improvement in the total cost. According to how the solution algorithms model the relationship between the locational and the routing sub-problems, Nagy and Salhi (2007) classified the heuristics as sequential, clustering, iterative, and hierarchical heuristics. The heuristic algorithm proposed here is a hierarchical heuristic in which the location problem is resolved as the main algorithm and the routing problem is solved as a subroutine in each step. The

proposed heuristic, which is made up of the drop, add and the swap mechanisms, is used at the location stage, and the sweep-based algorithm is applied at the routing stage (see Fig. 3). An improvement phase which includes a local search and a packing algorithm is used as a post-optimizer.

[Insert Figure 3 about here]

3.1 Add/Drop/Swap Algorithm

The add algorithm, initially proposed by Kuhn and Hamburger (1963), is a greedy heuristic algorithm which enlarges the set of used depots one by one through using the depot that yields the largest decrease in the objective function value. The algorithm stops when there is an increase in the total cost and the best solution is recorded. The Drop heuristic is then activated and attempts to drop one facility at a time until the cost starts rising. This is originally given by Feldman *et al.* (1966). Once the solution cannot be improved by dropping any of the facility, the add call is then applied and the shifting between the two schemes continues until there is no improvement. A swap-based move that attempts to simultaneously close and open a facility is then activated to see whether there is a better solution. This is achieved by using the fast interchange heuristic by Whitaker (1983). An even more efficient implementation that is useful for larger instances is presented by Resende and Werneck (2007). This incorporates data structure which records intermediate solutions and avoids re-computing unnecessary calculations.

The determination of a good solution at the initial location stage is important as it identifies a good set of open facilities without recourse to the routing elements. In other words, the problem reduces initially to solving the uncapacitated location problem. Once the routing component in subsection stages or iterations is integrated a limited call to the location stage will be sufficient. In this implementation the use of the swap move is restricted to a limited number of calls only to speed up the process. It was also observed that the add and the drop moves are used relatively less frequently.

The main steps of the algorithm are given below:

- Step 1:* Put all of the potential depots in List A, empty List B, and assign a big number as the initial total cost.
- Step 2:* Consider the depots in List B and the next potential depot in List A, assign the customers to their respective nearest depots, and calculate the total transportation cost based on the routes from the route construction. Repeat this step until all depots in List A have been tested.
- Step 3:* Obtain the potential depot that leads to the largest decrease in the total cost. If the largest decrease is positive, then move the corresponding potential depot which yields the minimum total cost from List A to List B and go to Step 2; otherwise go to Step 4.
- Step 4:* If the solution is not improved with the 'add' move go to Step 8; otherwise go

to Step 5.

Step 5: Drop the next depot in List B and consider the remaining ones. Calculate the total cost with the same approach in Step 2 and add the dropped depot in List B. Repeat this step until all depots in List B have been tested.

Step 6: Obtain the potential depot that leads to the largest decrease in the total cost. If the largest decrease is positive, then move the corresponding depot that yields the minimum total cost from List B to List A and go to Step 5; otherwise go to Step 7.

Step 7: If the solution is not improved with the ‘drop’ move go to Step 8; otherwise go to Step 2.

Step 8: Simultaneously open a closed depot and close an open depot to find the best combination yielding the cheapest total cost. If there is an improvement, update Lists A and B, and go to Step 2. If there is no improvement after k successive calls to the swap operator (in Fig. 3, we set $k=5$ as an example), record the solution and the search terminates.

3.2. Route Construction

Based on the pre-used depots and the assignment of customers to each depot, the sub-problem for each depot is reduced to a single depot vehicle routing problem with multi-compartment and multi-trip. The sweep algorithm, presented by Gillett and Miller (1974), is a simple but efficient heuristic to solve the routing problem. The simple implementation of the algorithm is described by Laporte *et al.* (2000). We propose a sweep-based algorithm by modifying the classical sweep method with the considerations of multiple compartments, multiple trips, and heterogeneous fleet.

In the sweep-based algorithm, note that each customer is a vertex and the depot is the square that is centered in the polar coordinate system. Assume that vertex j can be represented by its polar coordinates (θ_j, ρ_j) in which θ_j and ρ_j represent the angle and the ray length, respectively. Assign $\theta_j^* = 0$ to an arbitrary vertex j^* and compute the remaining angles, which are centered at 0 from the initial ray $(0, j^*)$. For a selected depot (e.g., depot i), we run the sweep method n_i times starting from n_i starting points where $n_i \in [1, |J_i|]$ in which J_i is the set of customers assigned to depot i , $i=1, 2, \dots, |I|$.

We then implement this algorithm on a depot by depot basis. The flowchart of our route construction is given in Fig. 4.

[Insert Figure 4 about here]

Rank the vehicles in a non-increasing order according to their fixed costs and test all of the vehicles as follows: first, if the type of vehicle v is the same as the previous vehicle, we skip vehicle v and test the next vehicle; second, we construct routes for the vehicle according to the sweep-based algorithm and calculate the following cost efficiency measure $\alpha_v = F_v(1 - u_v^c u_v^l)$ where u_v^c and u_v^l refer to the capacity and distance utilization of vehicle v , respectively. These are defined in (18) where l_{vt} and n_v^t are the total distance of trip t and the number of trips travelled by vehicle v , respectively. In

Equation (18), $\sum_j \sum_p D_{jp}$ represents the total demand for all types of products by the costumers served with vehicle v .

$$u_v^c = \frac{\sum_j \sum_p D_{jp}}{n_v^t \sum_p Q_{vp}} \quad \text{and} \quad u_v^l = \frac{\sum_{t=1}^{n_v^t} l_{vt}}{ML_v} \quad (18)$$

The vehicle with the lowest value for α_v is chosen to construct the route based on the sweep-based algorithm. This vehicle and its allocated customers are removed, and the process is repeated until there is no customer left to be served.

3.3. Packing-Strategy-based Local Search Procedure

An improvement procedure that combines a local search with a packing algorithm was applied to improve the best solution found thus far while the novel characteristics of multiple compartments and multiple trips are considered.

3.3.1. Packing Algorithm

The number of used vehicles in each depot could be reduced by using a packing algorithm with respect to pre-defined routes in the feasible solution. In this study, we propose a maximum remaining distance strategy in the packing algorithm to reduce the number of used vehicles in each depot. This procedure is similar to the one for the three-dimensional multi-container packing problem, the aim of which is to minimize the number of containers used to load all selected items (Feng *et al.* 2015). In the packing strategy, the feasible vehicles are firstly selected with respect to the capacity constraints and the longest remaining route is packed to the feasible vehicle with the maximum remaining distance. This procedure is as follows:

Step 1: Calculate the total distance of each route and rank the routes in decreasing order according to the corresponding distances.

Step 2: Rank the used vehicles in decreasing order according to the fixed costs.

Step 3: Remove the first vehicle that can both reduce the total cost and keep the solution feasible based on the packing strategy. If such a vehicle exists, remove the vehicle and repeat Step 3; otherwise stop.

3.3.2. The Local Search based on the Packing Strategy

The ‘exchange’ and ‘insertion’ operators are applied to both intra- and inter-routes, which are described as follows:

Insert: Within each route, move each customer to another position and calculate the savings. For inter-routes, move each customer from a route to another route while satisfying the capacities of the compartments and the maximum distance. Note that the two routes may depart from different depots yielding an inter-depot improvement.

Exchange: Within each route, swap two customers and calculate the new cost. For

inter-routes, exchange each customer from a route with a customer in the other route while considering the capacities of the compartments and the maximum travel distance. The inter-depot improvement, in which the latter route belongs to another depot, is also applied in this procedure.

The local search is run for a number of iterations and stops when no positive improvement is obtained. The algorithm adopts the first improvement strategy by accepting whichever operator yields such an improvement.

One disadvantage of the heuristic algorithm is that one customer's demands of all types of products are satisfied by one depot, which may limit the solution space. We develop an HGA that allows more flexible assignment of customers' demands of different types of products.

4. A Hybrid Genetic Algorithm

In recent years, the genetic algorithm (GA) that was developed by Holland in the 1960s has proved efficient in solving difficult optimization problems. In this paper, a hybrid genetic algorithm (HGA) is developed to solve the problem. In contrast to the constructive heuristic algorithm, the decisions of the depot selection and the assignment of the customers are obtained from the chromosomes in the HGA. A slightly different sweep-based algorithm is applied to construct the routes, and the same local search is embedded to improve the solutions obtained from the genetic operators. Moreover, the chromosomes allow the demands for different types of products by one customer to be assigned to different depots, which offers a wider space for route construction compared with that of the heuristic algorithm. This situation means that the solution obtained from the HGA may enable vehicles from depots in one area to finish the delivery task for vehicles with idle compartment capacities. The flowchart of the proposed HGA is shown in Fig. 5.

[Insert Figure 5 about here]

4.1. Chromosome Representation

In this paper, we encode the depot selection and the customer assignment into a single genetic code with two strings. In the chromosome, the first $|I|$ genes (depot genes) show the statement of the potential depots, and the remaining $|J| \times |P|$ genes (product genes) show the sequence of the products. The two strings for depot genes are the same, and the two strings for the product genes together imply the products ordered by the customers. If the position under product i is occupied by customer j , then the gene means that the type of the product is i and it is ordered by customer j . The largest positive number among the depot genes means that the product genes from the position to the end are assigned to the depot. The other positive numbers among the depot genes indicates that the product genes from the positive number to (the next larger number-1) are assigned to the depot. If there is only one remaining positive number, the remaining product genes are assigned to the depot. A depot will not be

used if the number occupying the depot is zero. Hence, the selection of the depot has also been obtained from the chromosome. Additionally, the arbitrary vertex i^* in the route construction for that depot is the starting vertex assigned to the depot. For example, a chromosome for a 4-depot, 8-customer and 2-product problem could be configured, as shown in Fig. 6.

[Insert Figure 6 about here]

From this chromosome, we can find that Depots 1, 3, and 4 are used. The largest positive number in the depot genes is 7, which means that the product genes from the 7th position to the end are assigned to Depot 3. The second largest positive number in the depot genes is 5, which means that the product genes from 5th position to the 6th (=7-1) position are assigned to Depot 1. The next largest positive number in the depot genes is 3 and it is the only remaining positive number, which means that the remaining product genes are assigned to Depot 4. In the route construction, the starting customer is 4 for Depot 1, 1 for Depot 3, and 5 for Depot 4. Note that one chromosome cannot present a complete solution. For a depot selection and customer assignment scenario determined by a chromosome, we use the route construction explained in the above section to obtain the routing solution.

4.2. Population Initialization

We generate an initial solution by using a heuristic method in the initialization process. The initialization mechanism is as follows:

Step 1: Randomly generate a subset of potential depots.

Step 2: Randomly generate a sequence of customers and assign each customer to the nearest depot in the subset.

Step 3: Assign the products to the customers in increasing order.

Step 4: Construct the chromosome based on the results from the first three steps.

4.3. Fitness function

The fitness function is used to evaluate the quality of a chromosome. Because the proposed mathematical model aims to minimize the total cost, we use the reciprocal of the objective function as the fitness function. Therefore, a chromosome is considered as a good one if it has a high fitness value.

4.4. Reproduction

In this paper, chromosomes with higher fitness values will be involved in the reproduction process with a higher probability. The Tournament Selection is used as the primary selection mechanism. First, we calculate the fitness value of each chromosome in the population. Then, two chromosomes are randomly selected from the population with replacement, and we sent the fitter one to the new population. The above two steps are repeated until the number of selected chromosomes is equal to the population size.

4.5. Crossover and mutation

In this paper, we use the two-cut-point method to obtain new sequences of products in the product genes, and use the one-cut-point method to the depot selection in the depot genes. We set a possibility, (i.e., crossover rate) to decide whether to conduct the crossover operator. If the crossover operator should be applied, then it is conducted with the following steps.

Step 1: Randomly select two different genes within the range of [1, length of the product genes] as crossing points, and exchange the product information presented by the two points.

Step 2: Move the product information of Parent 1 that are not selected from Parent 2 to Child 1 in the originating sequence.

Step 3: Apply the one-cut-point method to the depot genes.

Fig. 7 illustrates an example of crossover when $|I|=4$, $|J|=8$, and $|P|=2$.

[Insert Figure 7 about here]

We set a possibility, (i.e., mutation rate) to decide whether to conduct the mutation operator. We use different mechanisms for the depot genes and product genes. For the former part, we randomly select a position within the range of [1, length of the depot genes], and replace the position with a randomly generated number within the range of [0, length of the product genes]. For the latter part, we randomly select two different positions within the range of [1, length of the product genes] and reverse the products information between the two points (see Fig. 8).

[Insert Figure 8 about here]

4.6. Route Construction

The gene in our GA can tell the depot selection, the assignment of the customers to the depots, and the starting customer for each depot by using two strings. According to the information, we need to decide the route construction for one used depot and the customers assigned to it. We explore a sweep-based algorithm to determine the routes based on the chromosome obtained from the GA operators. The algorithm is similar to our constructive heuristic except the starting customer is selected by the chromosome and need not be generated. For cases in which some types of products for one customer are assigned to different depots, we considered the demands for the remaining types of products as zero to avoid the waste of compartment space during the vehicle selection process.

4.7. Mini Local Search based on the Packing Strategy

The HGA can generate a large number of customer assignment scenarios. On the other hand, however, the route construction might be relatively limited compared with the heuristic, because the starting customer is decided in each chromosome. Therefore, we design a mini local search to improve the best solution obtained in each generation to improve the effectiveness of the HGA. In the mini local search based on the

packing strategy, we focus on the intra-route exchanges and the ‘insert’ and ‘exchange’ moves are the same as those discussed in Section 3.3.

5. Computational Experiments

In this section, several numerical experiments are conducted to evaluate the performance and the efficiency of the proposed algorithms. The presented mixed integer program is implemented and solved by CPLEX (Version 12.1). All of the experiments are run on a personal computer with Intel (R) Core (TM) i3-2100 CPU at 3.10GHz, 4 G of RAM and Windows 7.

5.1 Experimental data design

In the problem, several parameters are given as input data, as mentioned in Section 2. In the experiments, we deal with two compartments in each vehicle because it is practical to classify the food as frozen and non-frozen. We consider four types of potential depots, the fixed costs of which are \$300, \$350, \$400, and \$500, respectively. Three types of vehicles are used in the experiments. From Type 1 to Type 3 of vehicles, (F_v, C_v, ML_v, Q_{vp}) are $(\$50, 2\$/km, 300km, (100,100))$, $(\$70, 3\$/km, 270km, (200,200))$, and $(\$100, 4\$/km, 250km, (300,300))$. The coordinates of the customers and potential depots are randomly generated in $U(0, 100)$, respectively. The demands of the two types of products are randomly generated in $U(50, 100)$.

5.2 Parameter setting

Four parameters are involved in the GA: the number of generations, the population size, the crossover rate, and the mutation rate. We conducted pilot tests to find the suitable values for these parameters. The 50-customer problem is used to achieve the effective tests, as shown in Experiment 3. We empirically set the initial values for each test prior to conducting rigorous testing. For the method of fine-tuning the population size, crossover rate, and mutation rate, refer to Moon *et al.* (2015).

Populations of 100, 200, 300, and 400 are tested as the potential population sizes. According to the results, we select 300 as the population size because of its high performance in the objective function value, computational time, and convergence rate. As Figs. 9 and 10 show, 0.7 and 0.2 are the best crossover and mutation rates to achieve a good objective function value, respectively.

[Insert Figures 9 and 10 about here]

5.3 Experiment 1 (optimal solution guaranteed)

In Experiment 1, small problems, including 8 customers, 2 potential depots, and 4 vehicles, are tested by using the mixed integer program and the proposed algorithms. The types of the two potential depots are Type 1 and Type 3. The numbers of each type of used vehicles in the experiments are 0, 2, and 2. Vehicles 1 and 2 belong to

Type 2 and Vehicles 3 and 4 belong to Type 3. All other information, including the demands and locations of customers and depots, is available on the website of the authors which will be notified later due to blind reviewing.

We first test one problem instance in order to observe the feasibility of each solution and compare the solutions obtained with the three approaches. In this instance, the randomly generated distance between the nodes and the demand of the customers are shown in Table 2. Pilot experiments using the VRP with Multiple Compartments (VRPMC) were conducted to verify the quality of the heuristic algorithm and the HGA. Each tested instance using the VRPMC involved 8 customers, one depot with a fixed cost of \$300, and 4 vehicles. The vehicles are of the same types as those used in Experiment 1 except that each vehicle can travel in only one trip. From Table 3, we can see that the two algorithms are efficient for the VRPMC.

In Table 4, the solutions including the objective function value, computational time, used depots, used vehicles, and the routes of each vehicle have been compared. We could observe that the results are the same except for the computation time.

[Insert Tables 2-4 about here]

Ten problem instances of the same characteristics as above are randomly generated and the objective function values compared in Table 5. In this particular experiment, both the constructive heuristic and the HGA obtain the optimal solutions which show their efficiency on these small size instances.

[Insert Table 5 about here]

5.4 Experiment 2 (optimal solution not guaranteed)

Four larger size problem instances are tested in which CPLEX failed to solve the mathematical model to optimality due to an out of memory error. We used instances with 20, 50, 75, and 100 customers. Therefore, the heuristic and the HGA are compared with each other. For each size of problems, ten instances with randomly generated distances and demands are used to conduct the experiments. Table 6 summarizes the results of these problems. The results of all of the ten instances for Experiments 2 are shown in the Appendix but a brief discussion on the results is given here for each of the four sizes which we refer to them as cases for simplicity.

[Insert Table 6 about here]

5.4.1. Case 1 (20 customers and 4 depots)

This experiment involves 20 customers, 4 depots, and 10 vehicles, the information of which are randomly generated. The numbers of each type of potential depot are 2, 0, 1, and 1 and the numbers of each type of available vehicles are 4, 3, and 3. Vehicles 1-4 belong to Type 1, Vehicles 5-7 belong to Type 2, and Vehicles 8-10 belong to Type 3.

The constructive heuristic obtained better solutions for 4 instances among the 10 instances, while the HGA found the best solution in 2 instances, and a tie was observed in the remaining 4 instances. On average the constructive heuristic produced a slightly smaller objective function value in a shorter average computation time amounting for about 60% of the HGA time.

5.4.2. Case 2 (50 customers and 4 depots)

Large problem instances containing 4 potential depots and 50 customers are randomly generated. The potential depots are the same as in Case 1. The numbers of each type of available vehicles are 10, 15, and 5 (a total of 30 vehicles). Vehicles 1-10 belong to Type 1, Vehicles 11-25 belong to Type 2, and Vehicles 25-30 belong to Type 3. The constructive heuristic obtained better solutions for 3 instances among the 10 instances, while the HGA found the best solution in 7 instances. On average the HGA produced a slightly smaller objective function value but required a longer computation time with an average of 1964.0 seconds compared to 924.0 seconds needed by its counterpart.

5.4.3. Case 3 (75 customers and 6 depots)

To further test the proposed algorithms, large problem instances containing 6 potential depots and 75 customers are randomly generated. The numbers of each type of potential depots are 2, 1, 2, and 1. Compared with Case 2, additional 5 vehicles of Type 3 are added. As shown in Table 6, neither of the two algorithms is dominated by the other. The HGA could obtain better solutions with a higher probability, i.e., the HGA could obtain better solutions for 8 instances among the 10 instances. The average computation time of the constructive heuristic is 1971.8 seconds whereas HGA averagely requires 2701.5 seconds.

5.4.4. Case 4 (100 customers and 8 depots)

To assess in more details case 3, we tested larger instances. Here, instances containing 8 potential depots and 100 customers are randomly generated. The numbers of each type of potential depots are 2, 2, 2, and 2. The vehicles are the same with those in Case 3. As shown in Table 6, the HGA could obtain better solutions for 9 instances among the 10 instances. For problems of this size, the average computation time of the constructive heuristic is 5524.8 seconds whereas the HGA consumes 5636.7 seconds.

From the above experiments, we can see that the HGA is the best performer when the number of customers is large and the performance gap between the two algorithms gets more significant when the problem size increases. Because the HGA can generate solutions in a wider space, the solution quality should be better than the heuristic algorithm at a cost of a longer computation time. For instance, for 100 customers, the HGA is able to outperform the counterpart algorithms in 90% of the time, but this

success comes with a much greater computational burden. Managers may select an algorithm according to the specific decision needed or their preference for a better solution or a shorter computational time.

5.5 Sensitivity Experiment

To obtain more managerial insights into the development of an efficient distribution system, we conducted sensitivity experiments on the density of node locations and the number of compartments. We set α as a coefficient to adjust the distances between nodes. We denoted the fixed cost of depots, fixed cost of vehicles, and transportation cost under α as $FCD(\alpha)$, $FCV(\alpha)$, and $TC(\alpha)$, respectively. We ran the HGA using an instance generated from Case 1, Experiment 2 by making more vehicles available. For a few fixed depots, a larger α means that the vehicles must travel longer distances between the depots and customers, which may be inefficient. Therefore, from the solutions, we found that more depots are used and the fixed cost of vehicles was also higher with a larger α (see Fig. 11). For example, when α is increased from 1.25 to 1.50, the fixed cost of the depots is increased from \$600 to \$1100. Consider $\alpha TC(1.00)$ as benchmarks and we can see that $TC(\alpha)$ is lower than $\alpha TC(1.00)$ when α is greater than 1.00, which implies a shorter traveling distance. Then, we conjectured that a trade-off characterizes the transportation cost with the fixed cost of the depots and the vehicles. For the Eastern Asian cities in China, Korea, and Japan with dense customer locations, a distribution system with a few big distribution centers (depot) might be efficient. In contrast, for European cities, such as those in the UK and France, more centers that cover a wide area might be more appropriate.

[Insert Figure 11 about here]

We conducted new experiments on five instances generated from Case 1, Experiment 2 by setting the number of compartments as 3, 4, and 5. Table 7 presents a summary of the total costs obtained from the heuristic algorithm and the HGA. Without a doubt, more compartments on one vehicle make the system more complex. In most cases, more compartments were associated with a higher total cost. In a few cases, however, more compartments may not lead to a higher system cost, such as when the existing solutions of depot location and vehicle management (including vehicle and route selection) can also satisfy the delivery demand for items in the new compartment(s). The heuristic algorithm and HGA may vary in terms of algorithm complexity and solution quality. Because the HGA has a greater flexibility for assigning demands for different types of products (i.e., different compartment requirement) to multiple vehicles, the quality of solutions obtained from the HGA is higher than those from the heuristics. However, the complexity of the HGA may also be higher than the heuristic algorithm because the HGA searches a wider solution space because of crossover and other mutation mechanisms.

[Insert Table 7 about here]

6. Conclusions and Suggestions

In this study, a location routing problem with multi-compartment and multi-trip (LRPMCMT) is investigated. This problem involves the selection of potential depots, the assignment of customers to depots, and the deployment of vehicles and routes for every vehicle as well as the assignment of products to the vehicles. The combined problem has never been addressed before, even though it is practical in real situations. To solve the small problems to optimality, a mixed integer program is presented. Because the problem is NP-hard, a heuristic and a hybrid genetic algorithm are also proposed to solve large problems. To evaluate the performance, the proposed algorithms are tested on randomly generated instances. Through a comparison with the optimal solutions from the mixed integer program, we demonstrated the efficiency of the algorithms. We could observe that neither of the proposed algorithms is dominated by the other when tested on the medium size problems (Case 2), however, the HGA obtains better solutions especially in larger instances though it needs a relatively longer computation time.

In future research, the mathematical formulation could be tightened by introducing valid inequalities as well as the lifting of some of the constraints. Note that in our packing heuristic, the possibility of customer reshuffling between the routes could lead to a better solution though this approach could be too time consuming given it is called several times within the search. However, its complexity, this is a challenging implementation that could be worth exploring further in the future.

From a practical view point, other extensions such as the inclusion of capacitated potential depots, the presence of time-window constraints and the need for both the delivery and the pickup considerations may be worth investigating. As this is an integrated problem spanning over different time scale which is made up of a strategic one (location decision) and tactical/operational (fleet size/ routing decision) a study that examines the robustness issue is in our view most exciting and very practical. In addition, because we allow each vehicle to conduct more than one trip in each day, the consideration of driver behaviors could be a fruitful direction. The uncertainty in the customer set could also be taken into account, making the optimization of the problem more complicated and challenging yet practically useful.

Acknowledgements

The authors are grateful to three anonymous referees for the useful comments. This work was supported by the by Social Science Foundation of Zhejiang Province (No. 17NDJC194YB). This work was partly supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning [Grant no. NRF-2017R1A2B2007812].

References

- Albareda-Sambola, M., Fernandez, E., Laporte, G., 2007. Heuristic and lower bound for a stochastic location-routing problem. *European Journal of Operational Research* 179 (3), 940-955.
- Azi, N., Gendreau, M., Potvin, J.Y., 2010. An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *European Journal of Operational Research* 202 (2), 756-763.
- Battarra, M., Monaci, M., Vigo, D., 2009. An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem. *Computers & Operations Research* 36 (11), 3041-3050.
- Berger, R., Coullard, R., Daskin, M., 2007. Location-routing problems with distance constraints. *Transportation Science* 41 (1), 29-43.
- Brandão, J., Mercer, A., 1997. A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. *European Journal of Operational Research* 100 (1), 180-91.
- Brown, G., Graves, G., 1981. Real-time dispatch of petroleum tank trucks. *Management Science* 27 (1), 19-32.
- Bruggen, L., Gruson, R., Salomon, M., 1995. Reconsidering the distribution structure of gasoline products for a large oil company. *European Journal of Operational Research* 81 (3), 460-473.
- Chajakis, E.D., Guignard, M., 2003. Scheduling deliveries in vehicles with multiple compartments. *Journal of Global Optimization* 26 (1), 43-78.
- Chen, J.X., Wang, S.A., Liu, Z.Y., Wang, W. 2017. Design of suburban bus route for airport access. *Transportmetrica A: Transport Science* 13(6), 568-589.
- Derigs, U., Gottlieb, J., Kalkoff, J., Piesche, M., Rothlauf, F., Vogel, U., 2011. Vehicle routing with compartments: applications, modeling and heuristics. *OR Spectrum* 33 (4), 885-914.
- Duhamel, C., Lacomme, P., Prins, C., Prodhon, C., 2010. A GRASP*ELS approach for the capacitated location-routing problem. *Computers & Operations Research* 37 (11), 1912-1923.
- Fallahi, A.E., Prins, C., Wolfler Calvo, R., 2008. A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers and Operations Research* 35 (5), 1725-1741.
- Feldman, E., Lehrer, F.A., Ray, T.L., 1966. Warehouse location under continuous economies of scale. *Management Science*, 12 (9), 670-684.
- Feng, X.H., Moon, I.K., Shin, J.H., 2015. Hybrid genetic algorithms for the three-dimensional multiple container packing problem. *Flexible Service and Manufacturing Journal*, 27, 451-477.
- Fleischmann, B., 1990. The vehicle routing problem with multiple use of vehicles. Working Paper, Fachbereich Wirtschaftswissenschaften, University Hamburg, Germany.
- Gillett, B.E., Miller, L.R., 1974. A heuristic algorithm for the vehicle dispatch problem. *Operations Research* 22 (2), 240-349.

- Kuhn, A.A., Hamburger, M.J., 1963. A heuristic program for locating warehouses, *Management Science* 9 (4), 643-666.
- Kulcar, T., 1996. Optimizing solid waste collection in Brussels. *European Journal of Operational Research* 90 (1), 26-44.
- Laporte, G., Gendreau, M., Potvin, J.Y., Semet, F., 2000. Classical and modern heuristics for the vehicle routing problem. *International Transactions in Operational Research* 7 (4-5), 285-300.
- Liu, P., Liao, F.X., Huang, H.J., Timmermans, H., 2016. Dynamic activity-travel assignment in multi-state supernetworks under transport and location capacity constraints. *Transportmetrica A: Transport Science* 12(7), 572-590.
- Mendoza, J.E., Castanier, B., Gueret, C., Medaglia, A.L., Velasco, N., 2010. A memetic algorithm for the multi-compartment vehicle routing problem with stochastic demands. *Computers & Operations Research* 37 (11), 1886-1898.
- Mendoza, J.E., Castanier, B., Gueret, C., Medaglia, A.L., Velasco, N., 2011. Constructive heuristic for the multi-compartment vehicle routing problem with stochastic demands. *Transportation Science* 45 (3), 346-363.
- Menezes, M.B.C., Ruiz-Hernández, D., Verter, V. 2016. A rough-cut approach for evaluating location-routing decisions via approximation algorithms. *Transportation Research Part B* 87, 89-106.
- Min, H., Jayaraman, V., Srivastava, R., 1998. Combined location-routing problems: a synthesis and future research directions. *European Journal of Operational Research* 108 (1), 1-15.
- Moon, I.K., Qiu, Z.B., Wang J.H. 2015. A combined tramp ship routing, fleet deployment, and network design problem. *Maritime Policy & Management* 42(1), 68-91.
- Muyldermans, L., Pang, G., 2010. On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm. *European Journal of Operational Research* 206 (1), 93-103.
- Nagy, G., Salhi, S., 2007. Location-routing: Issues, models and methods. *European Journal of Operational Research* 177 (2), 649-672.
- Olivera, A., Viera, O., 2007. Adaptive memory programming for the vehicle routing problem with multiple trips. *Computers & Operations Research* 34 (1), 28-47.
- Perl, J., 1983. A Unified Warehouse Location-Routing Analysis. PhD dissertation, Northwestern University.
- Petch, R.J., Salhi, S., 2004. A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Applied Mathematics* 133 (1-3), 69-92.
- Prins, C., 2002. Efficient heuristics for the heterogeneous fleet multitrip VRP with application to a large-scale real case. *Journal of Mathematical Modelling and Algorithms* 1 (2), 135-150.
- Prins, C., Prodhon, C., Wolfler Calvo, R., 2006, Solving the capacitated location routing problem by a GRASP complemented by a learning process and a path relinking. *4OR* 4 (3), 221-238.
- Prins, C., Prodhon, C., Ruiz, A., Soriano, P., Calvo, R.W., 2007. Solving the

- capacitated location routing problem by a cooperative lagrangean relaxation-Granular Tabu Search heuristic. *Transportation Science* 41 (4), 470-483.
- Prodhon, C. 2007. "An Iterative metaheuristic for the periodic location-routing problem." In *Operations Research Proceedings*, edited by S. Nickel, and J. Kalcsics, 159–164. Berlin: Springer.
- Prodhon, C., Prins, C., 2008. A memetic algorithm with population management (MA|PM) for the periodic location-routing problem. *Lecture Notes in Computer Science* 5296, 43-57.
- Prodhon, C., Prins, C., 2014. A survey of recent research on location-routing problems. *European Journal of Operational Research* 238, 1-17.
- Resende, M. G. C., Werneck, R., 2007. A fast swap-based local search procedure for location problems. *Annals of Operations Research* 150, 205-230.
- Salhi, S., Petch, R., 2007. A GA based heuristic for the vehicle routing problem with multiple trips. *Journal of Mathematical Modelling and Algorithms* 6(4), 591-613.
- Salhi, S., Rand, G.K., 1989. The effect of ignoring routes when locating depots. *European Journal of Operational Research* 39(2), 150-156.
- Silvestrin, P.V., Ritt, M., 2017. An iterated tabu search for the multi-compartment vehicle routing problem. *Computers & Operations Research* 81, 192-202.
- Taillard, E. D., G. Laporte, and M. Gendreau. 1996. "Vehicle routing with multiple use of vehicles." *Journal of Operational Research Society* 47: 1065–1070.
- Wang, S.A., Meng, Q., Yang, H. 2013. Global optimization methods for the discrete network design problem. *Transportation Research Part B* 50, 42-60.
- Watson-Gandy, C.D.T., Dohrn, P.J., 1973. Depot location with van salesmen-a practical approach. *Omega* 1(3), 321-329.
- Whitaker, R. A., 1983. A fast algorithm for the greedy interchange for large-scale clustering and median location problems. *INFOR* 21, 95-108.
- Yu, V.F., Lin, S.W., Lee, W., Ting, C.J., 2010. A simulated annealing heuristic for the capacitated location routing problem. *Computers & Industrial Engineering* 58 (2), 288-299.
- Zarandi, M.H.F., Hemmati, A., Davari, S., 2011. The multi-depot capacitated location-routing problem with fuzzy travel times. *Expert Systems with Applications* 38 (8), 10075-10084.

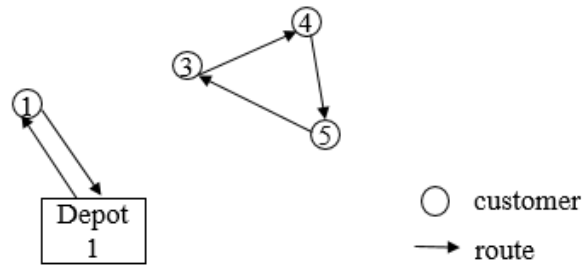
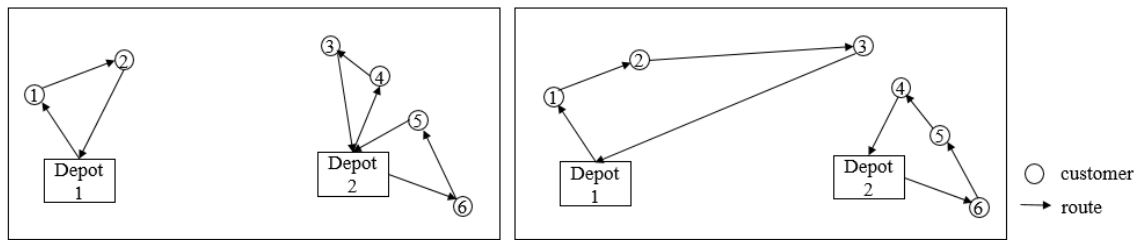


Fig. 1. An example of an infeasible solution due to subtour



(a) Customer assignment with the shortest distance strategy

(b) Customer assignment with vehicle utilization consideration

Fig. 2. Comparison of different customer assignments and route constructions

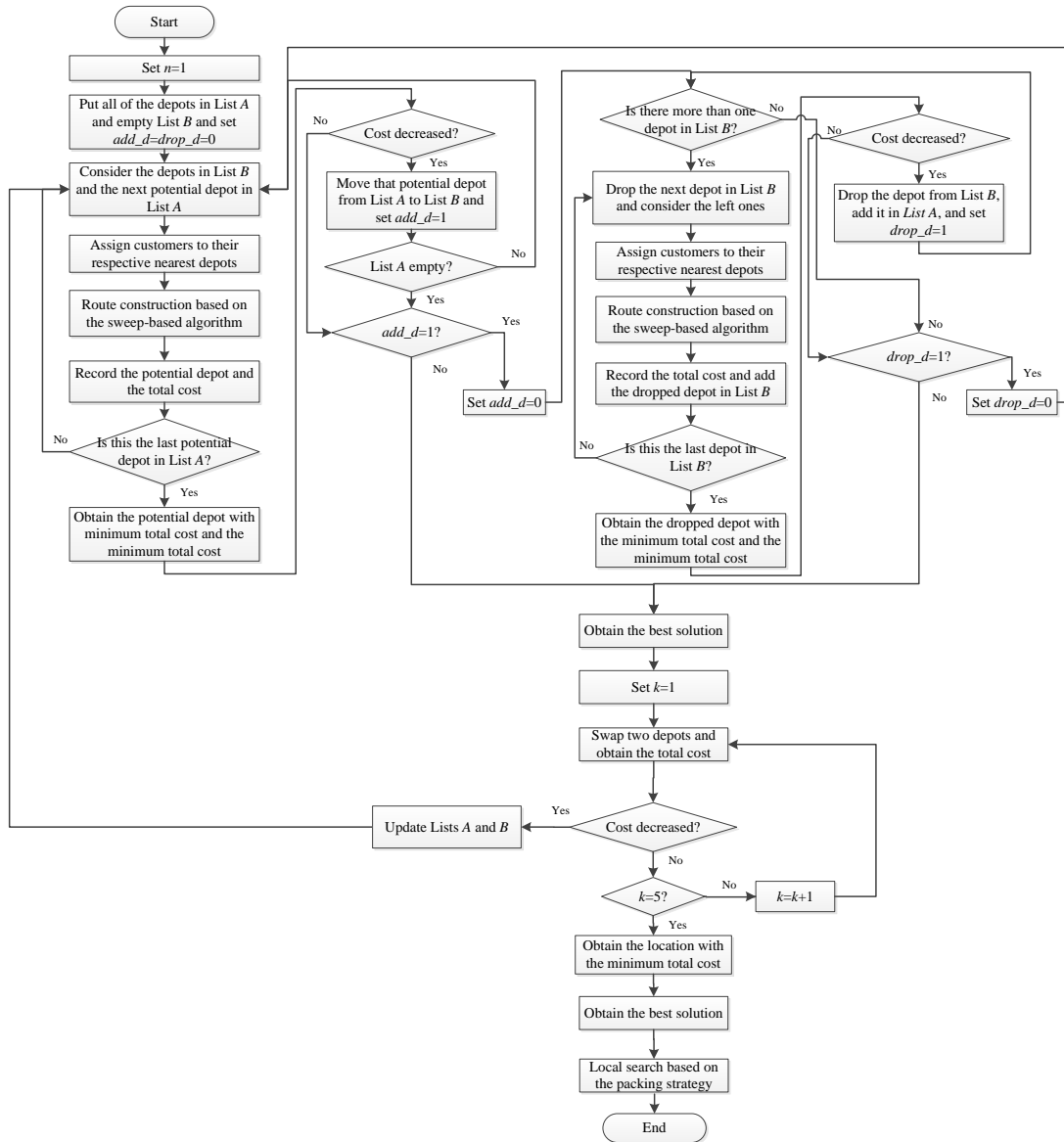


Fig. 3. Flowchart of the add/drop/swap heuristic

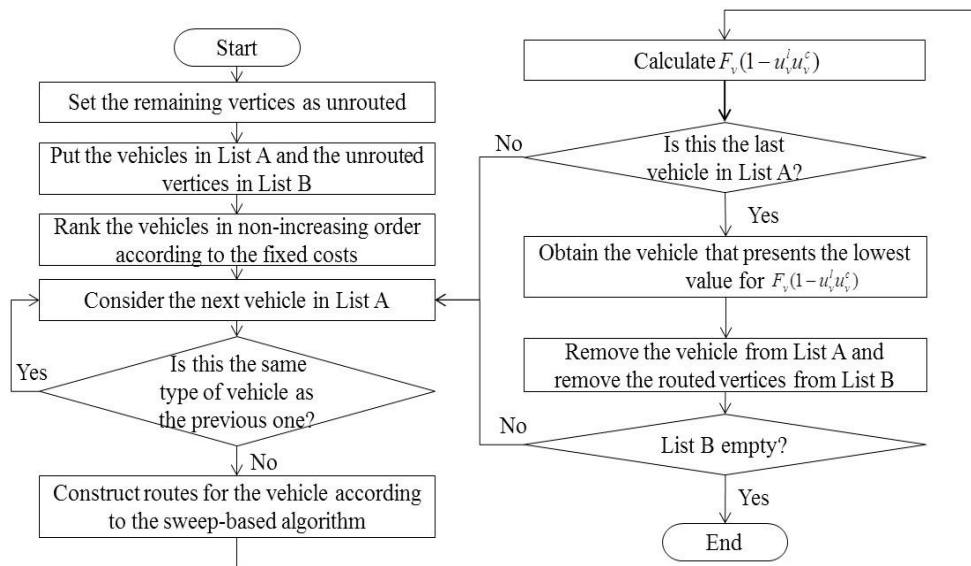


Fig. 4. Flowchart of the route construction

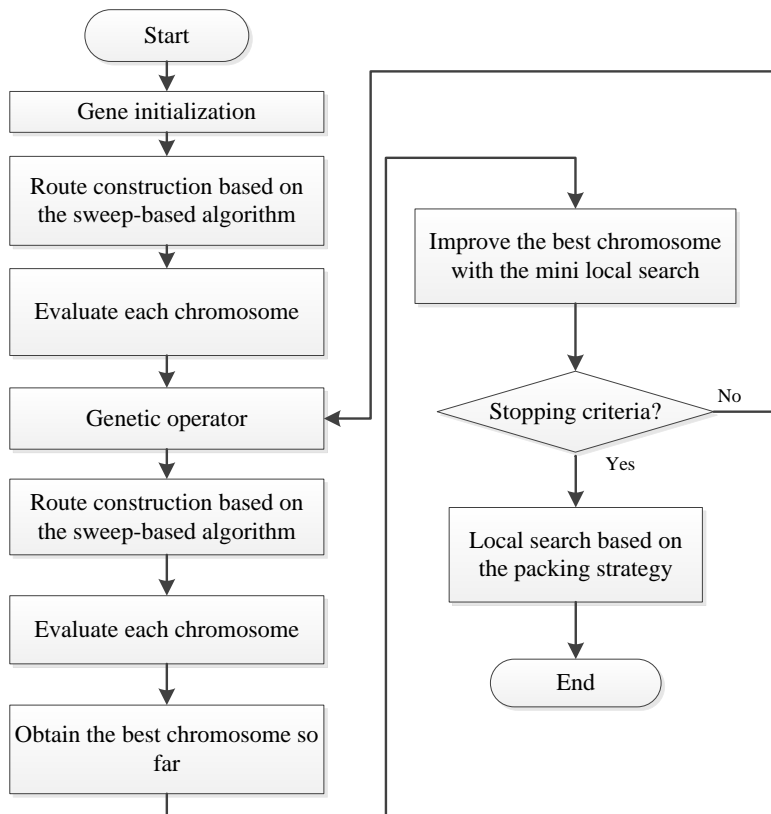


Fig. 5. Flowchart of the HGA

5	0	7	3	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	Product No.
5	0	7	3	5	5	3	3	4	4	1	1	7	7	8	8	2	2	6	6	Customer No.
Depot genes				Product genes																

Fig. 6. A possible chromosome for a sample problem

Parent 1	5	0	7	3	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	5	0	7	3	5	5	3	3	4	4	1	1	7	7	8	8	2	2	6	6
Parent 2	0	4	2	6	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	0	4	2	6	5	5	2	2	1	1	4	4	6	6	7	7	3	3	8	8
Child 1	5	0	7	6	1	2	1	1	2	2	1	2	1	2	1	2	1	2	1	2
	5	0	7	6	3	3	1	7	7	1	4	4	6	6	8	8	2	2	5	5
Child 2	0	4	2	3	1	2	1	1	2	2	1	2	1	2	1	2	1	2	1	2
	0	4	2	3	2	2	4	6	6	4	1	1	7	7	3	3	8	8	5	5

Fig. 7. Procedure for crossover

Parent 1	5	0	7	9	1	2	1	1	2	2	1	2	1	2	1	2	1	2	1	2
	5	0	7	9	3	3	1	7	7	1	4	4	6	6	8	8	2	2	5	5
Child 1	5	0	0	9	1	2	1	1	2	1	2	1	2	1	2	2	1	2	1	2
	5	0	0	9	3	3	1	7	8	8	6	6	4	4	1	7	2	2	5	5

Fig. 8. Procedure for mutation

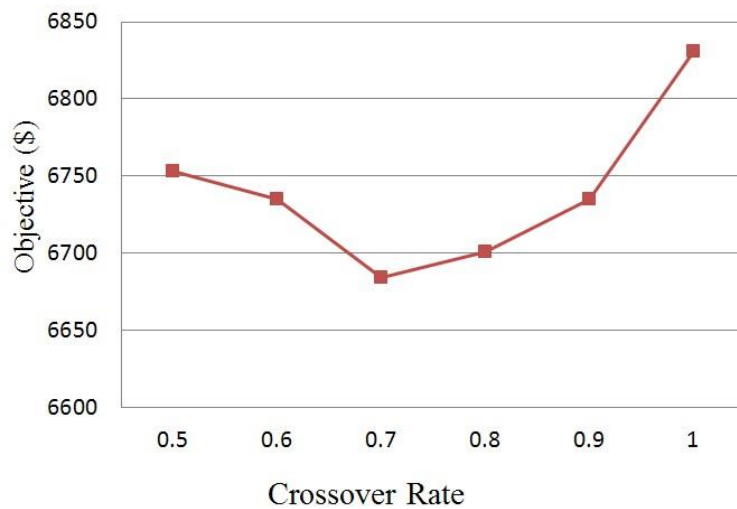


Fig. 9. Test results for crossover rates

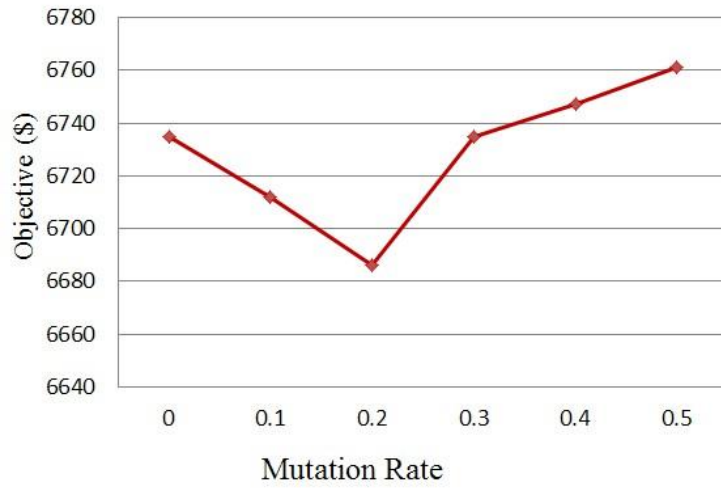


Fig. 10. Test results for mutation rates

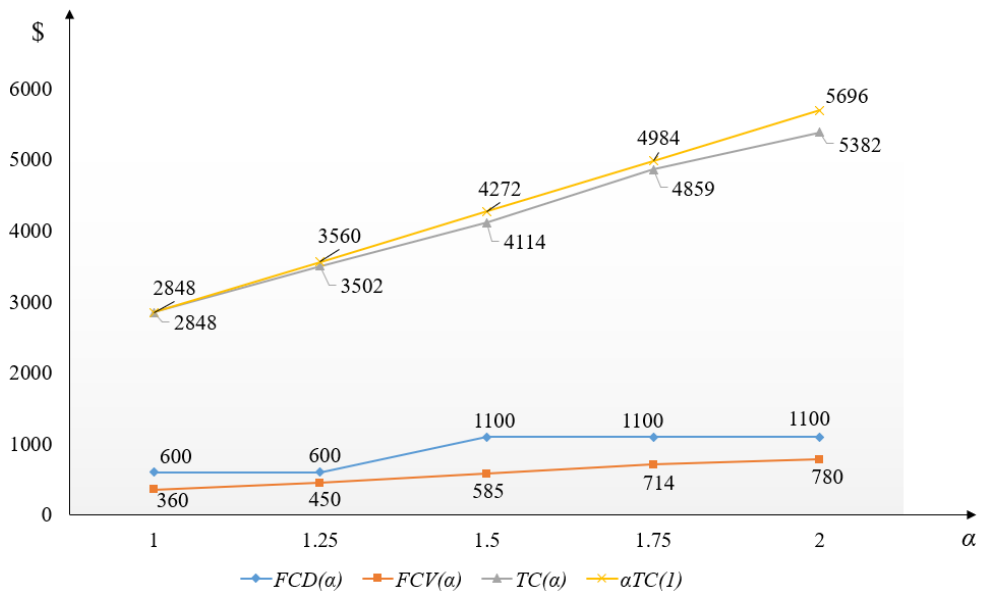


Fig. 11. Comparison of costs under different values of α

Table 1. Comparison of the problems

	LRP	VRPMC	VRPMT	This study
Depot	Multiple	Single	Single	Multiple
Compartment	Single	Multiple	Single	Multiple
Trip	Single	Single	Multiple	Multiple
Vehicle	Heterogeneous/ Homogeneous	Homogeneous	Heterogeneous/ Homogeneous	Heterogeneous

Table 2. Distance between nodes (km) and demand of customers (units)

		1	2	3	4	5	6	7	8	9	10
From	1	0	78	26	78	57	37	60	29	89	58
	2	78	0	58	23	28	44	56	79	44	53
	3	26	58	0	66	46	14	62	47	82	58
	4	78	23	66	0	21	54	38	71	21	37
	5	57	28	46	21	0	36	29	51	35	26
	6	37	44	14	54	36	0	58	52	72	54
	7	60	56	62	38	29	58	0	40	36	4
	8	29	79	47	71	51	52	40	0	75	39
	9	89	44	82	21	35	72	36	75	0	36
	10	58	53	58	37	26	54	4	39	36	0
Demand	Product 1	58	97	69	66	74	65	76	87		
	Product 2	65	65	93	89	56	50	87	56		

Note: Nodes 1 to 8 are the customers, while the other two nodes are potential depots.

Table 3. Results of pilot experiments for the MCVRP

Dataset	Objective value (\$)	MIP	Heuristic	HGA
		Time (seconds)	Time (seconds)	Time (seconds)
1	1427	1550.69	0.03	3.30
2	2131	1038.17	0.03	4.50
3	1912	441.67	0.02	4.00

4	1560	1791.36	0.02	3.80
5	2034	2785.34	0.04	4.20
6	1744	1360.33	0.01	3.90
7	2080	1703.05	0.03	3.70
8	1838	735.77	0.03	4.50
9	1710	527.63	0.03	4.40
10	1804	956.31	0.04	3.90

Table 4. Solutions obtained with the three approaches

		MIP	Heuristic	HGA
Objective function value (\$)		1734	1734	1734
Computation time (seconds)		8604	0.05	8.40
Used depots		10	10	10
Used vehicles		1,3	1,3	1,3
Vehicle 1	Trip 1	10-2-4-10	10-2-4-10	10-2-4-10
	Trip 2	10-7-5-10	10-7-5-10	10-7-5-10
Vehicle 2	Trip 1	10-6-3-1-8-10	10-6-3-1-8-10	10-6-3-1-8-10

Table 5. Comparison of the results obtained with the MIP and the proposed algorithms

Dataset	Objective value (\$)	MIP	Heuristic	HGA
		Time (hours)	Time (seconds)	Time (seconds)
1	1734	3.49	0.05	8.40
2	1566	2.18	0.07	8.30
3	1874	4.26	0.08	8.60
4	1704	3.15	0.05	8.90
5	1803	5.27	0.08	9.10
6	1941	4.94	0.06	8.90
7	1815	3.64	0.07	7.80
8	1438	3.11	0.07	9.60
9	1793	3.29	0.06	8.30
10	2240	4.78	0.07	8.60

Table 6. Results of Experiment 2

Problem		Objective value (\$)	
		Heuristic	HGA
20 customers and 4 depots	Average	3350	3350
	Average deviation	1.000	1.000
	Max deviation	1.042	1.014
	Best deviation	0.985	0.959
	# best	4	2
50 customers and 4 depots	Average	6065	6032
	Average deviation	1.005	0.995
	Max deviation	1.026	1.014
	Best deviation	0.986	0.975
	# best	3	7
75 customers and 6 depots	Average	7569	7493
	Average deviation	1.011	0.990
	Max deviation	1.029	1.015
	Best deviation	0.990	0.971
	# best	2	8
100 customers and 8 depots	Average	10131	9752
	Average deviation	1.039	0.963
	Max deviation	1.101	1.000
	Best deviation	1.000	0.909
	# best	0	9

Note: deviation=objective value of heuristic (HGA) /objective value of HGA (heuristic)

Table 7. Comparison of the results obtained with the proposed algorithms under different numbers of compartments

Dataset	2 compartments		3 compartments		4 compartments		5 compartments	
	Heu.	HGA	Heu.	HGA	Heu.	HGA	Heu.	HGA
1	3270	3021	3270	3021	3333	3119	3333	3119
2	3325	3257	3363	3339	3380	3344	3405	3353
3	3356	3344	3475	3449	3540	3449	3645	3604
4	3249	3258	3332	3301	3506	3489	3577	3525
5	3373	3237	3497	3497	3594	3501	3790	3578

Note: Heu. Represents Heuristic Algorithm and the unit of objective values is \$.

Appendix

See Tables 8 to 11.

Table 8. Detailed results of Experiment 2 with 20 customers and 4 depots

Dataset	Heuristic		HGA	
	Objective value (\$)	Time (seconds)	Objective value (\$)	Time (seconds)
1	3973	90.5	3973	125.1
2	3177	87.0	3190	141.4
3	3328	79.3	3369	147.7
4	3116	100.8	3115	139.0
5	3064	95.4	3064	142.2
6	3822	91.1	3878	145.4
7	3301	90.3	3301	150.0
8	3165	97.2	3201	139.4
9	3403	89.0	3265	147.0
10	3146	96.3	3146	146.4

Table 9. Detailed results of Experiment 2 with 50 customers and 4 depots

Dataset	Heuristic		HGA	
	Objective value (\$)	Time (seconds)	Objective value (\$)	Time (seconds)
1	5391	864.2	5337	1991.0
2	5885	928.5	5884	2189.5
3	6494	914.7	6353	1987.1
4	5722	1009.2	5702	1834.9
5	6437	934.0	6320	1877.3
6	6885	811.4	6712	2090.2
7	5559	872.4	5621	2053.6
8	6166	868.4	6143	1810.1
9	5863	1136.5	5919	1801.7
10	6243	900.6	6331	2004.2

Table 10. Detailed results of Experiment 2 with 75 customers and 6 depots

Dataset	Heuristic		HGA	
	Objective value (\$)	Time (seconds)	Objective value (\$)	Time (seconds)
1	7118	1850.9	7039	2661.0
2	6999	2050.1	6994	2770.1
3	7725	2110.4	7508	2597.8
4	6971	2023.6	6845	2800.7
5	7840	1829.4	7617	2455.2
6	7558	1986.9	7520	2666.6
7	7374	1997.6	7446	2709.1
8	8321	2167.6	8202	2871.0
9	7182	2045.3	7029	2588.1
10	8601	1656.6	8726	2895.6

Table 11. Detailed results of Experiment 2 with 100 customers and 8 depots

Dataset	Heuristic		HGA	
	Objective value (\$)	Time (seconds)	Objective value (\$)	Time (seconds)
1	10211	4501.2	10155	8333.1
2	9809	5112.0	9809	8090.0
3	10025	5765.4	9780	9263.4
4	9974	4879.1	9185	10010.5
5	10017	5381.5	9881	9989.7
6	10441	4793.2	9487	8762.3
7	10657	6166.7	10123	9776.8
8	10207	5083.9	9716	9015.5
9	9797	4987.6	9583	8224.3
10	10169	5577.8	9801	9003.0